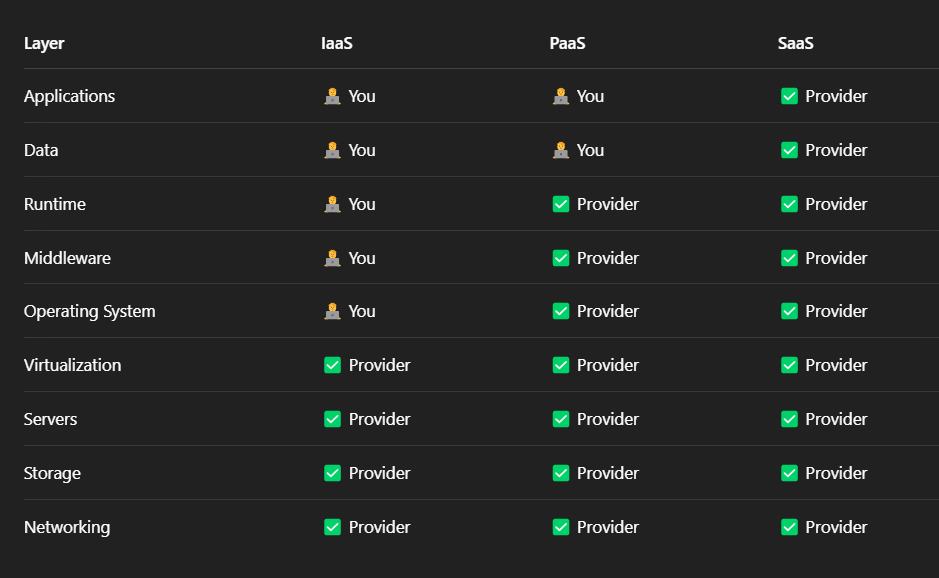
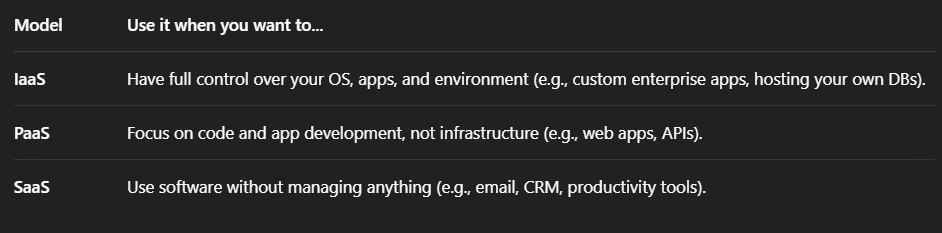
**Azure Cloud:**

* **What is Cloud?**
  + Cloud computing is the on-demand delivery of computing services—such as servers, storage, databases, networking, software, and analytics—over the internet.

**Azure cloud Services:**

* IaaS ( Infrastructure as a service )
* SaaS ( Software as a service )
* PaaS ( Platform as a Service )



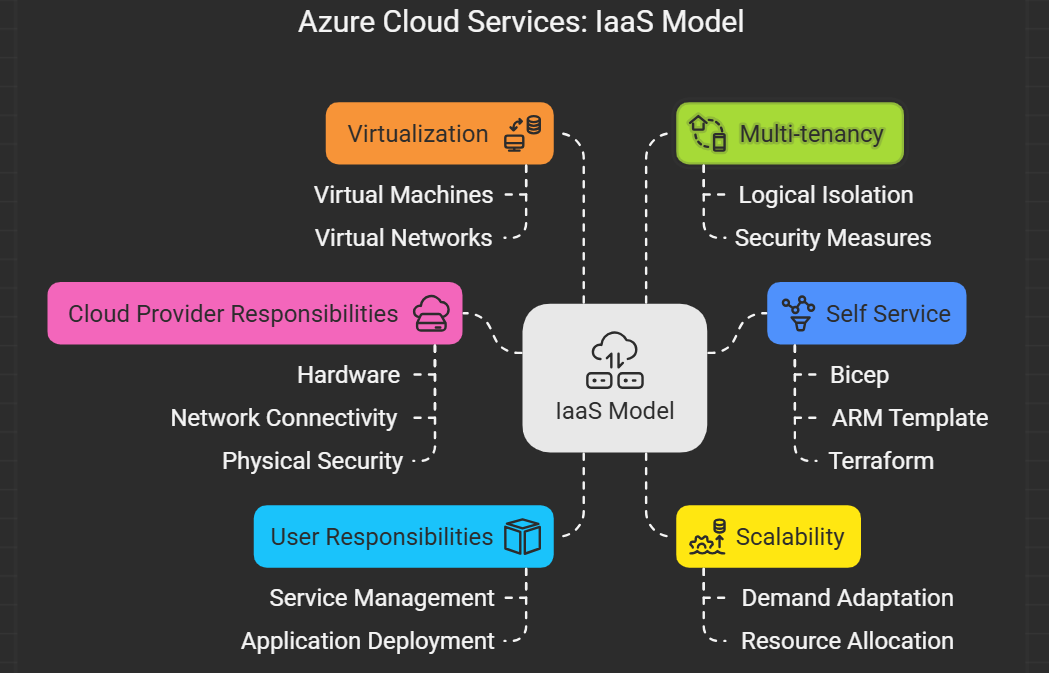


**IaaS**

* In an IaaS model, the cloud provider is responsible for maintaining the hardware, network connectivity (to the internet), and physical security.
* In the **Infrastructure as a Service (IaaS)** model, the cloud provider takes care of the **underlying physical infrastructure**, and **you (the user)** are responsible for managing almost everything that sits **on top of that infrastructure**.
* Scalability changing demands can be done.
* Self service ( bicet, arm template, terraform )

like we can create any service by using biceps & arm template & portal azure .com

* Virtualization
* Multi-tenancy ( sharing among multiple users while ensuring logical isolation and security for each tenant. )

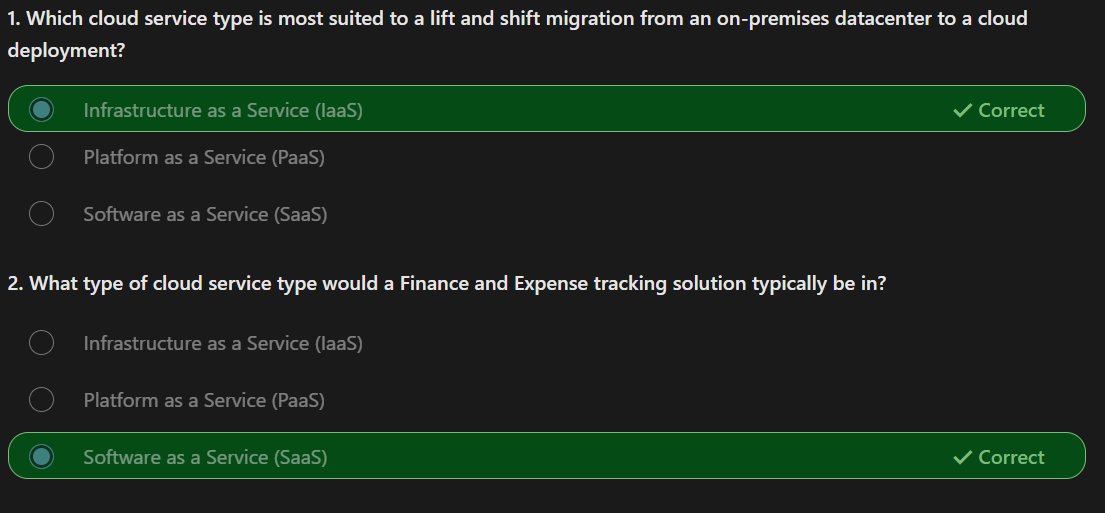


**PaaS**

* PaaS offers a platform allowing customers to develop, run, and manage applications without dealing with the underlying infrastructure.​
* It streamlines the development process, enabling developers to focus on application logic rather than infrastructure management.​

**SaaS**

* SaaS delivers software applications over the internet on a subscription basis.​
* It provides users with access to software without the need for installation or maintenance, reducing costs and complexity.​



* **Deployment Models?**
  + **Public Cloud**: Services are delivered over the public internet and shared across organizations.​
  + **Private Cloud**: Services are maintained on a private network, offering greater control and security.​
  + **Hybrid Cloud**: Combines public and private clouds, allowing data and applications to be shared between them.

Isolated:

* Isolation primarily refer to the separation of network traffic within and between Vnets.

Vnet:

* In a cloud provider network an Vnet is logically isolated section, where they can allow us to launch and manage (such as vm,etc).
* In Vnet only allowed resources will be running not other resources will come into it, it will only when we allow then into that Vnet.
* In Vent we have some address spaces that contains a range of Ip address where all the resources inside the Vnet will use. Each resource need an Unique IP address. So they can talk to each other.

#### 

**Step 1: Clarifying VNet vs. Subnets**

The VNet’s address space (10.0.0.0/16) is the entire backyard, with 65,536 IP addresses.

Subnets are smaller sections of that backyard, like little gardens, each with its own smaller range of IP addresses (e.g., 10.0.1.0/24, which has 256 addresses).

The /16 in 10.0.0.0/16 describes the VNet’s total range, not a subnet. Subnets typically use a higher number, like /24 or /28, to carve out smaller chunks.

**Quick Check:** Does it make sense that the VNet (10.0.0.0/16) is the whole backyard, and subnets are smaller pieces (like 10.0.1.0/24) inside it? If this is clear, we’ll move on to how subnets work. If not, let me know, and I’ll rephrase!

**Step 2: Why Do We Need Subnets?**

Imagine your backyard (VNet) is huge, and you want to organize it better. You don’t want all your toys (resources like web servers, databases, etc.) scattered everywhere. Instead, you create smaller sections (subnets) for different purposes.

Example:

One garden (subnet) for web servers (e.g., 10.0.1.0/24).

Another garden for databases (e.g., 10.0.2.0/24).

**Why?:**

Organization: Keeps things tidy—web servers in one place, databases in another.

Security: You can put a fence (security rules) around each garden to control who gets in or out. For example, only web servers can talk to databases, not the internet.

Efficiency: Makes it easier to manage lots of resources.

Question for you: Can you picture your VNet’s backyard with smaller gardens (subnets) for different types of resources, like one for web servers and one for databases? Does this idea of organizing and securing resources make sense? Let me know if you’re with me!

**Step 3: How Subnets Use the VNet’s Address Space**

Let’s go back to your VNet (10.0.0.0/16), which has 65,536 IP addresses. You decide to create two subnets:

Subnet A: For web servers, using 10.0.1.0/24 (256 IP addresses).

Subnet B: For databases, using 10.0.2.0/24 (256 IP addresses).

**What Does /24 Mean?**

The /24 in 10.0.1.0/24 means the first 24 bits of the IP address are fixed, and the last 8 bits can change.

Since 8 bits can make 2^8 = 256 combinations, this subnet has 256 IP addresses (from 10.0.1.0 to 10.0.1.255).

Same for Subnet B (10.0.2.0/24): another 256 addresses (10.0.2.0 to 10.0.2.255).

**What Happens to the VNet’s Addresses?**

Each subnet takes a chunk of the VNet’s 65,536 addresses.

Subnet A uses 256 addresses, Subnet B uses another 256, so together they use 512 addresses.

You still have 65,536 - 512 = 65,024 addresses left to create more subnets or assign to other resources.

**Fun Fact:** In real cloud setups (like Azure), a few IP addresses in each subnet are reserved (e.g., for network management), so you might get 251 usable addresses out of 256.